



Adverse Medical Event Prediction from Phone Calls

Team: TechIndians

(Divija Arora, Alok Kumar and Vinit)

DCRUST Batch of 2026

Veersa Hackathon 2026 : Case 1

GitHub : <https://github.com/itsalok2/TechIndians/>

Introduction

Healthcare systems handle millions of patient-nurse phone conversations daily.

Often, early signs of adverse medical events go unnoticed during these interactions, leading to delayed intervention and potential harm.

This project aims to **identify and predict possible adverse medical events from recorded phone conversations** by analyzing extracted symptoms and mapping them to historical adverse event data from the **FDA Adverse Event Reporting System (FAERS)**.

Problem Statement

The challenge is to detect indications of adverse medical events from patient-nurse phone calls, where symptoms may be described informally or indirectly.

Objectives:

- Transcribe recorded phone calls into text
- Extract medical symptoms using NLP techniques
- Map symptoms to known adverse events using FAERS data
- Assess severity and flag high-risk cases

Solution Overview

The proposed system integrates **speech processing, natural language processing, and real-world adverse event data** to proactively identify medical risks.

High-level workflow:

1. Audio input from patient–nurse phone calls
2. Speech-to-text transcription
3. Symptom extraction using NLP
4. Mapping symptoms to FAERS adverse events
5. Severity scoring based on FAERS outcomes
6. Flagging potential high-risk adverse events

Medical Audio Risk Detector (Prototype)

Upload an audio file, transcribe it, and translate it to English.

Upload Audio File



Drag and drop file here

Limit 200MB per file • MP3, WAV, M4A

[Browse files](#)

The user uploads the audio file

(Prototype)

Upload an audio file, transcribe it, and translate it to English.

Upload Audio File



Drag and drop file here

Limit 200MB per file • MP3, WAV, M4A

[Browse files](#)



Recording.m4a 277.3KB



0:00 / 0:12



Transcribing audio...



Transcribed Text (Detected Language):

Hello doctor, I took parasitamolam the morning because I had mild fever. Now I am ok but I have a little headache.

Audio is then Transcribed to text using Whisper



Transcribed Text (Detected Language):

Hello doctor, I took parasitamolam the morning because I had mild fever. Now I am ok but I have a little headache.



Translating to English...

GB English Translation:

Hello doctor, I took parasitamolam the morning because I had mild fever. Now I am ok but I have a little headache.



Extracting symptoms from English text...



Detected Symptoms:

headache, fever

The Symptoms and Drug taken are extracted

Data Source and Description

FAERS Dataset:

- Source: FDA Adverse Event Reporting System
- Link: <https://open.fda.gov/data/faers/>
- Year Used: **2021 (Q1–Q4)**
- Format: ASCII text files

System Architecture

The system follows a modular and scalable architecture.

Components:

- **Audio Processing Module:** Handles phone call recordings
- **Transcription Module:** Converts audio to text
- **NLP Module:** Extracts medical symptoms from text
- **FAERS Knowledge Base:** Stores historical adverse event data
- **Severity Analysis Module:** Computes risk level
- **Alerting Module:** Flags critical adverse events

Design Reference:

Complete architecture and UI flow are available via the Figma design link:

<https://cat-floral-16728185.figma.site/>

Data Processing Pipeline

Due to the large scale and normalized structure of FAERS data, careful preprocessing was required.

Key Steps:

- **Automated ingestion of ASCII files**
- **Column normalization and cleaning**
- **Filtering drugs marked as *Primary Suspect***
- **Aggregation of reactions and outcomes at the case level**
- **Severity score calculation using outcome codes**
- **Construction of a consolidated dataset for prediction**

This optimized pipeline avoids data explosion and ensures memory-efficient processing.

Quality Assurance (QA)

To ensure reliability and robustness, a structured QA approach was followed.

QA Activities:

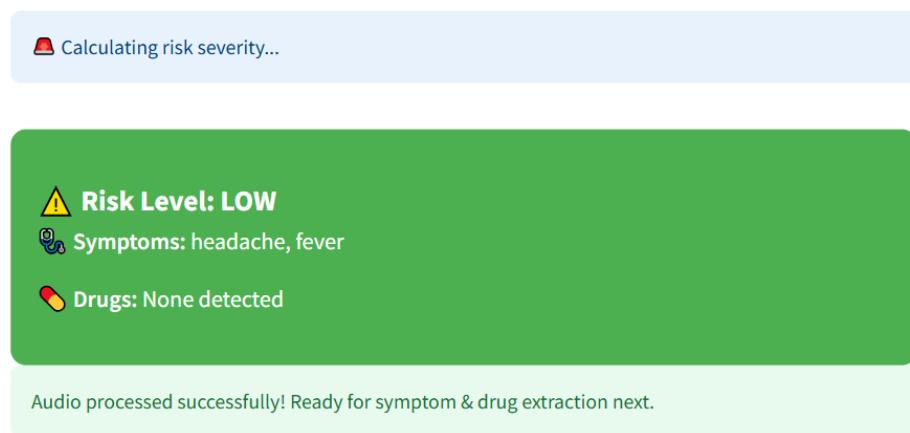
- ❑ Requirement analysis and documentation
- ❑ Manual test case execution
- ❑ Automated unit testing for data validation
- ❑ Verification of severity scoring and data integrity

Testing Approaches Used:

- ❑ Manual Test Cases
- ❑ Manual test case execution

Results and Observations

- Successfully processed large-scale FAERS data without memory issues
- Identified frequent adverse reactions and high-risk outcomes
- Enabled case-level severity analysis suitable for real-time applications
- Established a reliable pipeline for adverse event prediction.



The Result on the interfaces comes out as shown in the screenshot above

Conclusion

This project demonstrates how real-world adverse event data combined with NLP can help identify potential medical risks from phone conversations.

The solution is scalable, data-driven, and aligned with real healthcare challenges.

Future enhancements may include advanced NLP models, real-time deployment, and integration with clinical decision support systems.